Specification Amendments

Dated: 04/30/2010

Please replace the previous specification with the new one provided below.

Noise Filtering Edge Detectors

This application (following PCT/CA05/000017) is Continuation In Part of U.S Non-Provisional Application No. 10/520,040, filed on 27 December 2004; wherein such US 10/520,040, issued as US 7,564,934 on 21 July 2009, claims the benefits of PCT/CA2003/00909 filed on 25 June 2003 and claiming the benefits of Canadian Informal Application CA 2,389,969 filed on 25 June 2002.

BACKGROUND OF THE INVENTION

1. Field of the Invention

The parent application US 10/520,040 describes the DSP MSP invention, which includes noise filters for digital filtering of a captured waveform.

Such noise filters are shown;

in the parent application's Sec.3 of DESCRIPTION OF THE PREFERRED EMBODIMENT, and in this application GENERAL DESCRIPTION OF INVENTION COMPONENTS which repeats the Sec.2 of the SUMMARY OF THE INVENTION of the parent application.

This application contributes noise filters utilizing the method specified in the title as noise filtering edge detection. Such noise filtering edge detection offers fundamental advantages over conventional filters using the method of noise filtering amplitude detection (see the section 2). Therefore this invention represents major development of circuits and methods described in the parent application.

This invention defines digital means for programmable noise filtering from over-sampled

wave-forms consisting of variable lengths pulses having frequencies ranging from zero to 1/2 of technology's maximum clock frequency.

The noise filtering edge detectors (NFED) are directed to signal and data recovery in wireless, optical , or wireline transmission systems and measurement systems.

The noise filtering edge detectors (NFED) shall be particularly advantageous in system on chip (SOC) implementations of signal processing systems.

2. Background Art

Conventional noise filters calculate all output signal amplitudes corresponding to all digital sampling (or analog sensing) instances of input signals, in order to produce filtered output signals.

Since conventional filters spent their signal sampling (or sensing) resources and signal processing resources on calculating all reconstructed signal amplitudes, such conventional filters for serial links shall be named as amplitude noise filters.

Such amplitude filtering approach originated from AM domination in early communication era. It was appropriate one for data transmissions methods which use signal amplitudes as the main means for encoding transmitted data.

However; contemporary communication methods are based on FM, PM, or NRZ/PAM over copper/fiber which use signal transitions between limited set of discrete levels and transitions phases as the means for data encoding.

While conventional frequency domain signal processing is insufficient for identifying phase transients, prior art time domain signal processing requires by one order higher sampling rates and by several orders greater processing resources which cause it to be unaffordable for high speed data links.

Furthermore, conventional designs use frequency domain filters for recovering data from serially transmitted pulses. Since serially transmitted pulses must have widely variable lengths and frequencies, such frequency domain filters have to attenuate significant useful

part of such signal in order to eliminate high frequency phase jitter and high frequency amplitude glitches from such data carrying signal.

Conventional filters used in serial link receivers for optical/wireline/wireless communication, process received unfiltered signal in order to detect noise filtered amplitudes which construct an amplitude recovering signal.

Such conventional filters can be named noise filtering amplitude detectors, in order to differentiate them from this inventions noise filtering edge detectors.

Since transmitted data are carried by signal edges, said amplitude recovering signal produced by conventional filters is merely an intermediate signal which has to be processed further by an edge sensing circuit in order to recover information carried by signal edges which is necessary for actual data recovery.

Consequently; conventional receivers suffer from 2 inherent sources of errors, explained below:

1. Significant part of information needed for recovering signal edges has to be lost during the noise filtering amplitude detection, since conventional noise filtering amplitude detectors are unable to minimize edge phase noise as they have to minimize amplitude noise instead.
2. Since the amplitude recovering signal still has some amplitude noise in it and conventional edge sensing circuits can not provide any effective noise filtering, resulting prior art edge sensing introduces still more errors during said recovery of data carried by signal edges.

The above limitations of the conventional amplitude noise filters are alleviated by this invention's noise filtering edge detectors (NFEDs), as it is described below.

The parent application (US 10/520,040) allocates generic processing stages for noise filtering while designating close control and significant parts of noise filtering functions to be

performed by a Programmable Control Unit (PCU).

However the present application contributes more efficient and more specific noise filtering methods and hardware means than those enabled by US 10/520,040.

SUMMARY OF THE INVENTION

The NFED invention provides an implementation of programmable algorithms for noise filtering for a very wide range of low and high frequency wave-forms.

The NFED comprises; use of a synchronous sequential processor (SSP) for real time capturing and processing of in-coming wave-form, and use of a programmable computing unit (PCU) for controlling SSP operations and supporting adaptive noise filtering and edge detection algorithms.

General description of SSP and PCU configurations and operations is provided below in GENERAL DESCRIPTION OF INVENTION COMPONENTS.

Detailed description illustrating utilization of such SSP and PCU configuration for implementing the NFED is provided in DESCRIPTION OF THE PREFERRED EMBODIMENT in this application.

Even more comprehensive description of SSP circuits and timing control is provided in the parent application's Sec.3 of DESCRIPTION OF THE PREFERRED EMBODIMENT.

The NFED comprises using a set of binary values as an edge mask which is compared with a set of captured binary values surrounding a bit of a captured waveform buffer, in order to check if the captured bit represents an edge of the waveform.

Said comparison comprises:

- performing logical and/or arithmetic operations on particular bits of the edge mask and their counterparts from the waveform samples surrounding the particular bit of the

waveform buffer;

- performing arithmetic and/or logical operations on the results of said operations, in order to estimate waveform's edge proximity figure (EPF);
- comparing the EPF with an edge threshold, in order to determine if the captured bit represents an edge of the waveform.

The NFED further comprises modulating placement of detected rising and/or falling waveform edges by an edge modulating factor (EMF) calculated as a function of the EPF, were said function is controlled by an edge modulation control register (EMCR) which is preset by an external control unit.

The NFED still further comprises displacing detected rising and/or falling waveform edges by a preset number of bits, in order to compensate for [[ISI's]] Inter-Symbol-Interference (ISI) and/or other duty cycle distortions.

The NFED invention further includes:

- using the WFSC for incoming waveform registration and monitoring (see the GENERAL DESCRIPTION OF INVENTION COMPONENTS);
- programmable waveform analysis and adaptive noise filtering algorithms;
- edge mask registers for providing said edge masks used for detecting rising and/or falling waveform edges;
- edge threshold registers for providing said edge thresholds used for detecting rising and/or falling waveform edges;
- edge displacement registers for providing said edge displacement numbers used for shifting detected rising and/or falling edges by a programmable number of bits of waveform processing registers;
- filter control registers which control; said logical and/or arithmetic operations conducting the comparison of captured waveform bits with the edge mask, and said edge displacements in the processed waveforms;
- using the PCU for calculating and loading said edge mask registers and/or said edge

threshold registers and/or said edge displacement registers and/or said filter control registers;

- using the PCU for controlling said calculations of the EMF by presetting the EMCR in accordance with adaptive noise filtering algorithms.
- using the PCU for controlling and using the WFSC operations for implementing adaptive filters by controlling noise filtering edge detection stages of the SSP.

Such NFED comprises methods, systems and circuits described below.

1. A noise filtering edge detector (NFED) for removing phase noise from wave-form edges and/or removing amplitude glitches from wave-form pulses by continues digital filtering of the entire incoming wave-form sampled in time instances matching single gate delays provided by outputs of a delay line built with serially connected gates which a sampling clock is propagated through, wherein variable lengths pulses having frequencies ranging from zero to 1/2 of technology's maximum clock frequency are processed by comparing an edge mask, which provides an expected pattern of wave-form samples corresponding to an edge of the wave-form, with a sequence of wave-form samples surrounding a consecutive analyzed sample; the NFED comprising:

a wave capturing circuit for capturing results of sampling the incoming wave-form in time instances produced by the outputs of the delay line which the sampling clock is propagated through;

an apparatus for performing logical or arithmetic operations on particular samples of the edge mask and their counterparts from the wave-form samples surrounding the consecutive analyzed sample of the captured wave-form;

an apparatus for using the results of said operations for deciding if said operations can determine a filtered location of an edge of a filtered wave-form.

2. An NFED as described in statement 1, wherein said edge mask samples of the expected edge pattern are compared with samples from a consecutive processed region of the captured wave-form wherein correlation between a consecutive edge mask sample and a

corresponding sample from the processed region is estimated by performing an arithmetical or logical operation on said consecutive mask sample and on said corresponding sample from the processed region; the NFED comprising:

a circuit for accessing any said consecutive processed region of the captured wave-form and using such region as comprising samples corresponding to the edge mask samples;

a circuit for selection of a consecutive sample from the edge mask and for simultaneous selection of a corresponding consecutive sample from the processed region of the captured wave-form;

a circuit for calculating a correlation component between such selected samples by performing an arithmetical or logical operation on said selected samples;

a circuit for calculating a digital correlation integral by adding said correlation components calculated for single samples of the edge mask.

3. An NFED as described in statement 2, wherein said correlation integrals are calculated for said consecutive processed regions uniformly spread over all the captured wave-form wherein said calculated correlation integrals are further analyzed and locations of their maximums or minimums are used to produce said filtered locations of said edges of the filtered wave-form; the NFED comprising:

a circuit for moving said processed region by a programmable number of samples positions of the captured wave-form;

a circuit for storing and comparison of said correlation integrals calculated for different processed regions, in order to identify said maximums or minimums and their locations;

a circuit for using said locations of said maximums or minimums for producing the filtered locations of the edges of the filtered wave-from.

4. An NFED as described in statement 3, wherein noise is filtered and said storing and comparison of said correlation integrals are simplified by subtracting an edge threshold from any newly calculated correlation integral first and by disregarding all resulting decreased integrals if they are negative while using only positive decreased integrals for further noise filtering; the NFED further comprising:

a circuit for subtracting the edge threshold from any newly calculated correlation integral, in order to determine if such decreased integral indicates signal change greater than noise levels and to reduce amount of further processing;

an apparatus for dismissing those said decreased integrals which have negative values, and for classifying only those said decreased integrals which are still positive for a further signal processing including said comparisons.

5. An NFED as described in statement 1, wherein the NFED further comprises:

a filter arithmometer for comparing the edge mask with the captured wave-form in order to introduce noise filtering corrections of the edges of the filtered wave-form;

a filter mask register providing the edge mask which is compared with the captured wave-form of an input signal and/or filter control register which provides code for controlling operations of said filter arithmometer in order to provide said corrections of the filtered wave-form.

6. A noise filtering edge detector (NFED) as described in statement 1, wherein the NFED includes compensation of inter-symbol interference (ISI) or other predictable noise by adding a programmable displacement to said filtered location of the edge of the wave-form; the NFED comprising:

a circuit for programmable amendment of the filtered location of the wave-form edge by presetting said programmable displacement with a new content;

a circuit for using such newly preset displacement for shifting the filtered location of the next detected edge.

7. A noise filtering edge detector (NFED) as described in statement 1, wherein the NFED uses a set of binary values as the edge mask which is compared with a set of captured binary values surrounding the analyzed sample of the captured wave-form in order to produce an edge proximity figure (EPF) estimating a proximity of the analyzed sample to a nearest wave-form edge wherein the EPF is further compared with an edge threshold in order to detect if the analyzed sample can point out location of an edge of the filtered

wave-form; the NFED comprising:

a circuit for using the results of said operations for producing the edge proximity figure (EPF) estimating a mismatch between said nearest edge and the wave-form region surrounding the analyzed sample;

a circuit for comparing the EPF with the edge threshold, in order to determine if the analyzed sample provides said location of an edge of the filtered wave-form.

8. A noise filtering edge detector (NFED) as described in statement 7, wherein the NFED further includes compensation of periodical predictable noise with programmable modulations of said filtered locations of the wave-form edges by using an edge modulating factor (EMF) for a periodical diversification of said edge thresholds corresponding to different said regions of the wave-form; the NFED comprising::

a circuit for modulation of the filtered locations of the wave-form edges by using the edge modulating factor (EMF) for modulating said edge thresholds which are used for the evaluation of the EPF's calculated for said different wave-form regions surrounding different consecutive samples of the captured wave-form;

whereby said EMF provides such modulation of the edge thresholds, that predictable noise introduced to consecutive wave-form samples by known external or internal sources, is compensated.

9. A noise filtering edge detector (NFED) as described in statement 8, wherein:

said modulation of the edge thresholds is controlled by an edge modulation control register (EMCR) which is preset by an external control unit.

10. An NFED as described in statement 1, wherein the NFED comprises:

sequential processing stages configured into a sequential synchronous pipeline driven synchronously with said sampling clock.

11. An NFED as described in statement 10, further comprising parallel processing phases implemented with said synchronous sequential pipelines; wherein:

said parallel processing phases are driven by clocks having two or more times lower frequencies than said sampling clock;

consecutive parallel phases are driven by clocks which are shifted in time by one or more periods of said sampling clock;

12. An NFED as described in statement 11, wherein:

said wave-form filtering is extended beyond a boundary of a single phase by using multiple noise filtering sequential stages in every parallel processing phase.

13. An NFED as described in statement 12, including an over-sampled capturing of consecutive wave-form phases in corresponding phases wave registers which are further rewritten to wave buffers with overlaps which are sufficient for providing all wave samples needed for a uniform filtering of any edge detection despite crossing boundaries of the wave buffers which are loaded and used during different said phases; the NFED comprising:

a circuit for rewriting the entire wave register belonging to one phase into the wave buffer of the same phase and for rewriting an end part of said wave register into a front part of the next phase wave buffer, while the remaining part of the next wave buffer is loaded from the wave register belonging to the next phase;

whereby every wave buffer contains entire said wave-form regions needed for calculating said EPF's corresponding to the samples belonging to the phase covered by this buffer.

14. An NFED as described in statement 12, wherein:

carry over bit or bits of an output register of a first filter stage of one phase is or are clocked-in into an output register of the first filter stage of a next phase together with filtering results of the next phase;

a second filter stage of the next phase uses the output register of the first filter stage for filtering a wave-form interval which extends into the next phase.

15. An NFED as described in statement 12, comprising:

a circuit for merging of said parallel processing phases, wherein multiple said parallel processing phases are merged into a smaller number of parallel phases or into a single processing phase, when passing from one said sequential processing stage to the next sequential stage.

16. An NFED as described in statement 12, comprising:

a circuit for splitting of said parallel processing phases, wherein one said processing phase is split into multiple parallel processing phases or multiple parallel processing phases are split into even more parallel phases, when passing from one said sequential processing stage to the next sequential stage.

17. An NFED as described in statement 12, further including a programmable control unit (PCU) for reading results of captured signal processing from the NFED and for controlling operations of the NFED; wherein the PCU comprises:

a circuit for reading results of captured signal processing from the NFED;

an apparatus for programming the filter mask register and/or the filter control register and/or said presetting of the programmable displacement and/or the edge modulating factor, which are applied for achieving said filtering of the captured wave-forms.

18. An NFED as described in statement 1, further including a programmable control unit (PCU) for reading results of captured signal processing and for controlling operations of said signal processing.

19. An NFED as described in statement 1, further including a wave-form screening and capturing circuit (WFSC) for incoming waveform registration and monitoring wherein the WFSC identifies characteristics of the incoming wave-form captured with the resolution matching single gate delays; wherein the WFSC comprises:

a circuit for using programmable screening masks and/or programmable control codes for verifying incoming wave-form captures for compliance with said programmable

screening masks.

20. An NFED as described in statement 19, wherein the WFSC comprises:

a circuit for buffering captured wave-form for which the pre-programmed compliance or non-compliance has been detected, or for counting a number of said detections;

a circuit for communicating said buffered wave-form and/or a detections counter, to an internal control circuit and/or to an external unit.

21. An NFED as described in statement 20 further including a programmable control unit (PCU) for reading results of captured signal processing from the WFSC and for controlling operations of the WFSC; wherein the PCU comprises:

a circuit for programming the screening masks and/or the control codes for performing said verification of captured wave-forms compliance or non-compliance with said screening patterns;

a circuit for reading verification results and/or reading captured wave-forms which correspond to the preprogrammed verification criteria.

22. An NFED as described in statement 21 including implementation of adaptive noise filtering algorithms; wherein the PCU comprises:

means for programmable waveform analysis;

a circuit for loading edge mask registers which provide said edge masks used for detecting rising and/or falling wave-form edges;

or a circuit for loading edge threshold registers which provide said edge thresholds used for detecting rising and/or falling waveform edges;

or a circuit for loading edge displacement registers which provide said edge displacements used for shifting detected rising and/or falling edges by a programmable number of samples positions of the captured wave-form;

or a circuit for loading filter control registers which control said logical and/or arithmetic operations conducting the comparison of captured wave-form samples with the edge mask, and said edge displacements in the processed wave-forms;

or an apparatus for controlling said EMF by presetting the EMCR in accordance with adaptive noise filtering algorithms.

BRIEF DESCRIPTION OF THE DRAWINGS

1. Introduction

The drawings are self-explanatory as far as NFED circuit level implementation is concerned, since the clear drawing conventions are consistently applied as it is explained below.

The drawings are numbered correspondingly to processed data flow.
Interconnect signals between the the drawings have unique names identifying their sources and destinations explained in the Description of the Preferred Embodiment utilizing the same names.
Single-interconnect signals are drawn with a thin line, while multi-interconnect signals are drawn with a thick line.
Inputs supplied from different drawings are connected at the top or left side and outputs are generated on the bottom due to the top-down or left-right data flow observed generally.
Clocked circuits like registers or flip-flops are drawn with two times thicker lines than combinatorial circuits like arithmometers or selectors.

3. Brief Description of the Drawings of NFED

FIG.1 shows circuits for Wave Capturing including rewriting of an end part of one wave register together with a content of other wave register into the other register's buffer .
FIG.2 shows circuits for Sequential Clocks Generation for the NFED.
FIG.3 shows circuits for comparing captured signal samples with an edge mask and detecting a filtered edge location.

GENERAL DESCRIPTION OF INVENTION COMPONENTS

The DSP MSP (originated in the PCT/CA03/000909) provides an implementation of programmable algorithms for analyzing a very wide range of low and high frequency wave-forms.

The DSP MSP comprises:

a synchronous sequential processor (SSP) for real time capturing and processing of in-coming wave-form and a programmable computing unit (PCU) for controlling SSP operations and supporting adaptive signal analysis algorithms.

Said SSP comprises a multi_sampled phase (MSP) capturing of incoming wave-form level by a locally generated sampling clock and its sub-clocks generated by the outputs of serially connected gates which the sampling clock is propagated through. If an active edge of the wave-form is detected by capturing a change in a wave-form level, the position of the captured signal change represents an edge skew between the wave-form edge and an edge of the sampling clock.

In addition to the above wave-form capturing method, the SSP includes 3 other methods of the edge skew capturing which are defined below:

- the sampling clock captures the outputs of serially connected gates which the incoming wave-form is propagated through;
- the outputs of serially connected gates which the incoming wave-form is propagated through, provide wave-form sub-clocks which capture the sampling clock.
- the incoming wave-form captures the outputs of serially connected gates which the sampling clock is propagated through;

The above mentioned edge skew capturing methods further include:

- using falling edges of said sub-clocks for driving clock selectors which select parallel processing phases during which positive sub-clocks are enabled to perform said edge skew capturing, or using rising edges of said sub-clocks for driving selectors which select parallel processing phases during which negative sub-clocks are enabled to perform said edge skew capturing;
- using serially connected clock selectors for enabling consecutive sub-clocks, in order to assure that consecutive sub-clocks will target appropriate consecutive bits of appropriate capture registers.

The SSP ~~invention~~ includes using said serially connected gates:

- as being an open ended delay line;
- or being connected into a ring oscillator which can be controlled in a PLL

configuration;

- or being connected into a delay line which can be controlled in a delay locked loop (DLL) configuration.

Every said edge skew amounts to a fraction of a sampling clock period.

The SSP comprises measuring time intervals between active wave form edges, as being composed of said edge skew of a front edge of the incoming waveform, an integer number of sampling clock periods between the front edge and an end edge, and said edge skew of the end edge of the wave-form.

The SSP further comprises a parallel multiphase processing of incoming signal by assigning consecutive parallel phases for the capturing of edge skews and/or processing of other incoming wave-form data with clocks which correspond to consecutive sampling clocks.

Consequently the SSP comprises using 1 to N parallel phases which are assigned for processing incoming signal data with clocks corresponding to sampling clock periods number 1 to N, as it is further described below:

- circuits of phase1 process edge skews or phase skews or other incoming signal data with a clock which corresponds to the sampling clock period number 1;
- circuits of phase2 process edge skews or phase skews or other incoming signal data with a clock which corresponds to the sampling clock period number 2;
- finally circuits of phaseN process edge skews or phase skews or other incoming signal data with a clock which corresponds to the sampling clock period number N.

Said parallel multiphase processing allows N times longer capturing and/or processing times for said multiphase stages, compared with a single phase solution.

The SSP includes parallel stage processing of incoming signal by providing multiple processing stages which are driven by the same clock which is applied simultaneously to inputs of output registers of all the parallel stages.

The SSP further comprises a synchronous sequential processing of incoming signal by using multiple serially connected processing stages with every stage being fed by data from the

previous stage which are clocked-in by a clock which is synchronous with the sampling clock.

Since every consecutive stage is driven by a clock which is synchronous to the same sampling clock, all the stages are driven by clocks which are mutually synchronous but may have some constant phase displacements versus each other.

The SSP further comprises:

- merging of processing phases which occurs if multiple parallel processing phases are merged into a smaller number of parallel phases or into a single processing phase, when passing from a one processing stage to a next processing stage;
- splitting of processing phases which occurs if one processing phase is split into multiple processing phases or multiple processing stages are split into even more processing stages, when passing from a one processing stage to a next processing stage.

The SSP includes a sequential clock generation (SCG) circuit which uses said clock selectors and said sub-clocks: to generate SSP clocks which drive said parallel phases and said sequential stages, and to generate selector switching signals for said merging and splitting of processing phases.

The SSP includes time sharing of said parallel phases: which is based on assigning a task of processing of a newly began wave-form pulse to a next available parallel processing phase.

The SSP comprises a sequential phase control (SPC) circuit, which uses results of a wave edge decoding and said SSP clocks, for performing said time sharing phase assignments and for further control of operations of an already assigned phase.

The SSP comprises passing outputs of a one parallel phase to a next parallel phase, in order to use said passed outputs for processing conducted by a following stage of the next parallel phase.

The outputs passing is performed: by re-timing output register bits of the one phase by clocking them into an output register of the next parallel phase simultaneously with processing results of the next parallel phase.

The SSP further comprises all the possible combinations of the above defined: parallel multiphase processing, parallel stage processing, synchronous sequential processing, merging of processing phases, splitting of processing phases, and outputs passing.

The SSP includes processing stage configurations using selectors, arithmometers, and output registers, which are arranged as it is defined below:

- input selectors select constant values or outputs of previous stages or outputs of parallel stages or an output of the same stage to provide arithmometer inputs, and arithmometer output is clocked-in to an output register by a clock which is synchronous to the sampling clock;
- multiple arithmometers are fed with constant values or outputs of previous stages or outputs of parallel stages or an output of the same stage, and an output selector selects an arithmometer output to be clocked-in to an output register by a clock synchronous to the sampling clock;
- the above defined configuration as being supplemented by using an output of an output selector of a parallel processing stage for controlling output selector functions.

Proper arrangements of said parallel and sequential combinations and said stages configurations provide real time processing capabilities for very wide ranges of signal frequencies and enable a wide coverage of very diversified application areas.

The DSP MSP comprises two different methods for accommodating a phase skew between the sampling clock and a clock which drives the incoming wave-form, and both methods allow elimination of ambiguities and errors in decoding incoming signal data patterns. Said two methods are further defined below:

- a clock synthesizer is used for producing the sampling clock, which will maintain frequency or phase alignment with the clock which drives the incoming wave-form;
- expected phase skews between the sampling clock and the wave-form driving clock are calculated and used to correct measurements and data patterns decoding of the incoming signal pulses;
- both above mentioned methods include measurements of phase or frequency deviations of the incoming signal clock versus the sampling clock, and using said measurements results to control the clock synthesizer or to calculate the expected phase skews.

The DSP MSP comprises a fractional bit staffing (FBS) which improves accuracy of fixed point arithmetic far beyond of what conventional solutions could offer.

The FBS uses processing arguments which are expressed as a series of terms, where each term may have a differently staffed last bit or several last bits. Said differently staffed last bits express a fractional value which is combined with previous bits which express a constant more significant part of a processing argument.

The DSP MSP cumulative processing operations are split into a series of basic addition or subtraction or comparison operations. Every said consecutive term, of a processing argument of a cumulative operation, is used for processing performed during a corresponding consecutive basic operation.

Consequently using the FBS enables reducing of a total error of a long cumulative processing operation to a single last bit resolution.

The DSP MSP comprises: using phase differences between incoming signal pulses identified with the MSP captures and expected data patterns defined by sampling clock periods, for processing of the incoming signal and for detecting data patterns delivered by incoming signal pulses.

The DSP MSP further comprises more conventional method, which calculates whole time intervals of incoming signal pulses and divides them by time intervals of expected data patterns which would be defined in sampling clock periods.

It shall be noted however: that said use of the phase differences, which are small fractions of

the whole intervals, allows significant reductions in processing time and in processing hardware.

The DSP MSP includes noise filters for digital filtering of a captured wave-form, which include the circuits listed below:

- filter mask registers and filter control registers which can be downloaded from the PCU;
- digital filter arithmometers which use the mask registers content for correcting captured wave-forms in accordance with control codes provided by said filter control registers;

Said noise filters further include adding a second noise filter stage in every noise filtering parallel phase for the purpose of extending a range of a filtered waveform beyond a boundary of a single phase.

Said second filter stages shall have the same basic circuits as the above mentioned first filter stages.

In order to allow said boundary extension, carry over bit or bits of an output register of said first filter stage of one phase shall be clocked-in into an output register of the first filter stage of a next phase together with filtering results of the next phase. Consequently the second filter stage of the next phase shall use the output register of the first stage for filtering a wave-form interval which extends through both said phases.

The DSP MSP includes phase processing stages (PPS), which can perform listed below operations:

- calculating the front edge skew and the end edge skew of the in-coming wave-form pulses;
- combining said edge skews with other indicators of pulse duration and phase deviations between the sampling clock and a clock which generates the incoming wave-form;
- evaluating the resulting timing of the incoming wave-form pulses versus expected timing values which correspond to interpretation patterns of the incoming signal;
- communicating results of the above mentioned operations to other circuits of the DSP MSP.

The DSP MSP includes periodical skew accumulation (PSA) circuits, which can perform listed below operations:

- providing amounts of phase skew between the sampling clock period versus an expected period of a clock which drives the incoming signal;
- providing said phase skews as being different for every individual sampling clock period;
- reading the next set of said phase skews from external circuits, and seamless attaching them to a present set of the phase skews;
- calculating accumulations of said phase skews for every pulse or for combinations of pulses of the incoming signal;
- synchronous communicating of said accumulations of the pulse skews to the phase processing stages and/or to other parts of the DSP MSP.

The DSP MSP further includes received data collection (RDC) circuits for performing the operations, which are listed below:

- using outputs of the PPS and the PSA circuits for detecting and registering incoming data patterns;
- seamless combining of the registered data patterns into unified data blocks consisting of fixed number of bits or bytes;
- seamless communicating of the unified data blocks to an internal control unit and/or to an external unit.

The DSP MSP comprises data frequency capturing (DFC) circuits, for providing listed below operations:

- using outputs of MSP processing for detecting and registering time intervals of incoming signal pulses defined by the incoming signal clock;
- using outputs of RDC circuits for detecting and registering time intervals of the data patterns defined by the sampling clock, which correspond to the above mentioned

already registered time intervals of incoming signal pulses;

- seamless combining of the pulses time intervals bounded with the corresponding data patterns intervals into frequency measurement blocks providing relation between a frequency of the incoming signal clock and a frequency of the sampling clock;
- seamless communicating of the frequency measurement blocks to an internal control unit and/or to an external unit.

The DSP MSP comprises wave-form screening and capturing circuits (WFSC), for providing listed below operations:

- using programmable data masks and programmable control codes for verifying incoming MSP captures for compliance or non-compliance with a pre-programmed screening patterns;
- buffering captured data for which the pre-programmed compliance or non-compliance have been detected;
- counting a number of the above mentioned detections;
- communicating both the buffered captured data and the number of detections, to an internal control unit and/or to an external unit;
- using programmable time slot selection circuits for selecting a time interval for which incoming MSP captures shall be buffered and communicated to an internal control unit and/or to an external unit.

Said PCU comprises implementation of the functions listed below:

- programming of verification functions and patterns for checking captured wave-forms for compliance or non-compliance with the patterns;
- reading verification results and reading captured wave-forms which correspond to the preprogrammed verification criteria;
- reading captured wave-forms which can be pre-selected by the PCU arbitrarily and/or based on other inputs from the SSP;
- programming of noise filtering functions and noise filtering masks for filtering captured

wave-forms;

- reading results of real-time wave-form processing from the SSP, processing the results and providing control codes and parameters for further real-time wave-form processing in the SSP, in accordance with adaptive signal processing algorithms;
- reading output data from the SSP, interpreting the data, and communicating the data to external units.

The DSP MSP comprises said SDR MSP circuits , which further include listed below features:

- multisampling of every individual bit of a received data stream;
- detection and registration of a phase of every rising or falling edge of the received data stream;
- filtering out received signal noise;
- using lengths of a pulse of the incoming signal for calculating a number of data bits received in the pulse;
- registration of the detected data bits and seamless communication of the received data to an internal control unit and/or to an external unit.

DESCRIPTION OF THE PREFERRED EMBODIMENT

This embodiment implements the general components of the SSP and the NFED defined above, as it is shown in FIG.1, FIG.2 and FIG.3.

Consequently, this description specifies NFED detailed circuit implementation and NFED entire configuration and operations.

Multiple components shown in FIG.1 / FIG.2 / FIG.3 of the present application are also described in the parent application, as they are similar to the components shown in FIG.1 / FIG.2A / FIG.3A of the parent application.

Said NFED comprises the multi-sampled phase (MSP) capturing of incoming wave-form intervals in specifically dedicated wave interval registers which are further rewritten to wave interval buffers (see the FIG.1 showing the wave registers 1WR,2WR followed by the wave buffers 11WB, 12WB, 21WB, 22WB).

In order to provide all wave samples needed for the filtering edge detection along a whole wave buffer, the NFED invention includes rewriting:

- the end part 2WR(R:(R-M+1) of the wave register 2WR, into the front parts 11WB(M:1),12WB(M:1) of the wave buffers 11WB,12WB;
- the end part 1WR(R:(R-M+1) of the wave register 1WR, into the front parts 21WB(M:1),22WB(M:1) of the wave buffers 21WB,22WB.

The preferred embodiment is based on the assumptions listed below:

- the wave registers 1WR and the 2WR are 15bit registers (i.e. R=14);
- the rising edge mask REM(M:0) and the falling edge mask FEM(M:0) are 8bit registers (i.e. M=7) and the PCU loads the same masks equal to 00001111 to both mask registers;
- the rising edge threshold RET is loaded with 0110 (6 decimal), and the falling edge threshold FET is loaded with 0010 (2 decimal);

The digital filter arithmometers 21DFA1/22DFA1/11DFA1/12DFA1 perform all the comparison functions, between the edge mask registers REM/FEM and the waveform buffers 21WB/22WB/11WB/12WB involving the edge threshold registers RET/FET, with the 3 basic operations which are further explained below.

The first operation is performed on all the waveform bits and involves the edge mask bits as it is specified below:

For every waveform's consecutive bit WB_k the surrounding bits WB_{k-4} , WB_{k-3} , WB_{k-2} , WB_{k-1} , WB_k , WB_{k+1} , WB_{k+2} , WB_{k+3} are logically compared with the mask bits B_0 , B_1 , B_2 , B_3 , B_4 , B_5 , B_6 , B_M and the resulting 8bit binary expression $BE_k(7:0)$is created as equal to;

$BE_k(0) = (WB_{k-4}=B_0)$, $BE_k(1) = (WB_{k-3}=B_1)$, $BE_k(2) = (WB_{k-2}=B_2)$,

$BE_k(3) = (WB_{k-1}=B_3)$, $BE_k(4) = (WB_k=B_4)$, $BE_k(5) = (WB_{k+1}=B_5)$,

$BE_k(6) = (WB_{k+2}=B_6)$, $BE_k(7) = (WB_{k+3}=B_7)$.

The second operation adds arithmetically all the bits of the binary expression $BE_k(7:0)$ and the resulting edge proximity figure EPF_k is calculated as equal to $EPF_k = BE_k(0) + BE_k(1) + BE_k(2) + BE_k(3) + BE_k(4) + BE_k(5) + BE_k(6) + BE_k(7)$ which shall amount to a 0 - 8 decimal number.

The third operation performs functions explained below:

- The verification is made if the EPF_k indicates a rising edge condition by exceeding the content of the rising edge threshold RET(T:0). Consequent detection of the $EPF_k > RET = 6$ condition, sets to level = 1 the corresponding $DFR1_k$ bit of the DFR1 and all the remaining bits of the present DFR1 until a falling edge is detected as it explained below.
- The verification is made if the EPF_k indicates a falling edge condition by being smaller than the content of the falling edge threshold FET(T:0). Consequent detection of the $EPF_k < RET = 2$ condition, sets to level = 0 the corresponding $DFR1_k$ bit of the DFR1 and all the remaining bits of the present DFR1 unless a rising edge is detected as it explained above.

In order to carry the same level from the last bit of the previous phase DFR1 into the following bits of the present phase digital filter register2 (DFR2), the last bit DFR1(R) of the previous DFR1 is rewritten into the carry bit DFR1(C) of the present DFR1 and is used by the digital filter arithmometer2 (DFRA2) to fill front bits of the DFR2 with the same level as the last bit of the previous phase DFR1.

The digital filter arithmometers 21DFA2/22DFA2/11DFA2/12DFA2 perform; the inter-phase continuation of filling front bits of the present phase register in accordance with the level set in the last bit of the previous phase, followed by said edge displacement which compensates for duty cycle distortions due to ISIs, etc..

The edge displacement comprises the 3 basic operations described below.

- Any DFR1 rising edge, indicated by a level 0 to 1 transition, is shifted left by a number of bits specified by a content of the rising edge displacement register (RED(D:0)) loaded by the PCU in accordance with its filtering algorithms.
- Any DFR1 falling edge, indicated by a level 1 to 0 transition, is shifted left by a number of bits specified by a content of the falling edge displacement register (FED(D:0)) loaded by the PCU in accordance with its filtering algorithms.
- In order to propagate said displacement operations from the present processing phase to the previous processing phase; the propagated sign of the edge bit (DFR2(Sp)) and the propagated bits (DFR2(Dp:0)), are calculated by the DFA2 and are written down into the DFR2 extension DFR2(Sp,Dp:0).

In order to propagate said displacement operations from the next phase DFR2 into end bits of the present phase digital filter register3 (DFR3); the propagated sign of the edge bit and the propagated displaced bits DFR2(Sp,Dp:0) from the next phase, are used by the digital filter arithmometer3 (DFRA3) to fill end bits of the digital filter register 3 (DFR3) with the correctly displaced bits propagated from the next phase to the present phase.

The circuits which follow the DFR3, starting with the 2:1 SEL and ending on the 1Front

EdgeBuf. and 1EndEdgeBuf., are shown in FIG.1 to merely illustrate NFED integration with the data recovery system of the parent application. Since they belong to the phase processing stages of the parent application, they are described in Sec.3 of DESCRIPTION OF THE PREFERRED EMBODIMENT of the parent application and shown in its FIG. 3A.

Furthermore it is shown in the FIG.1, FIG.2, FIG.3; that the timing and circuits for further phase processing and data recovery can remain similar to those shown in the FIG.2D and FIG.3A, FIG.3B, FIG.3C of the parent application.

The obvious timing difference, is caused by increasing the number of noise filtering stages from 1 (shown in the FIG.3A) to 3 (shown in the FIG.1). Such timing difference can be easily accommodated by increasing the follow-up numbering of next stages clocks. More specifically starting from the Clk2 clock numbering shall be increased by 3; i.e. the 1Clk2 shall be replaced by the 1Clk5, and so on.

The detailed description of NFED utilization for the phase processing and data recovery specified in the parent application, have been provided above for explaining NFED operations only and in no way define, limit, construe or describe the scope or extent of NFED applications.